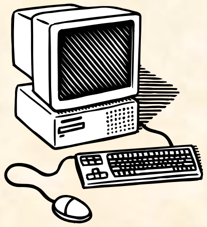
A Multi-Agent Architecture for Streaming Text Analysis

Joel Reed and Tom Potok
Applied Software Engineering Research Group
Computational Sciences and Engineering Division
Oak Ridge National Laboratory

Text Analysis Challenge

Vs



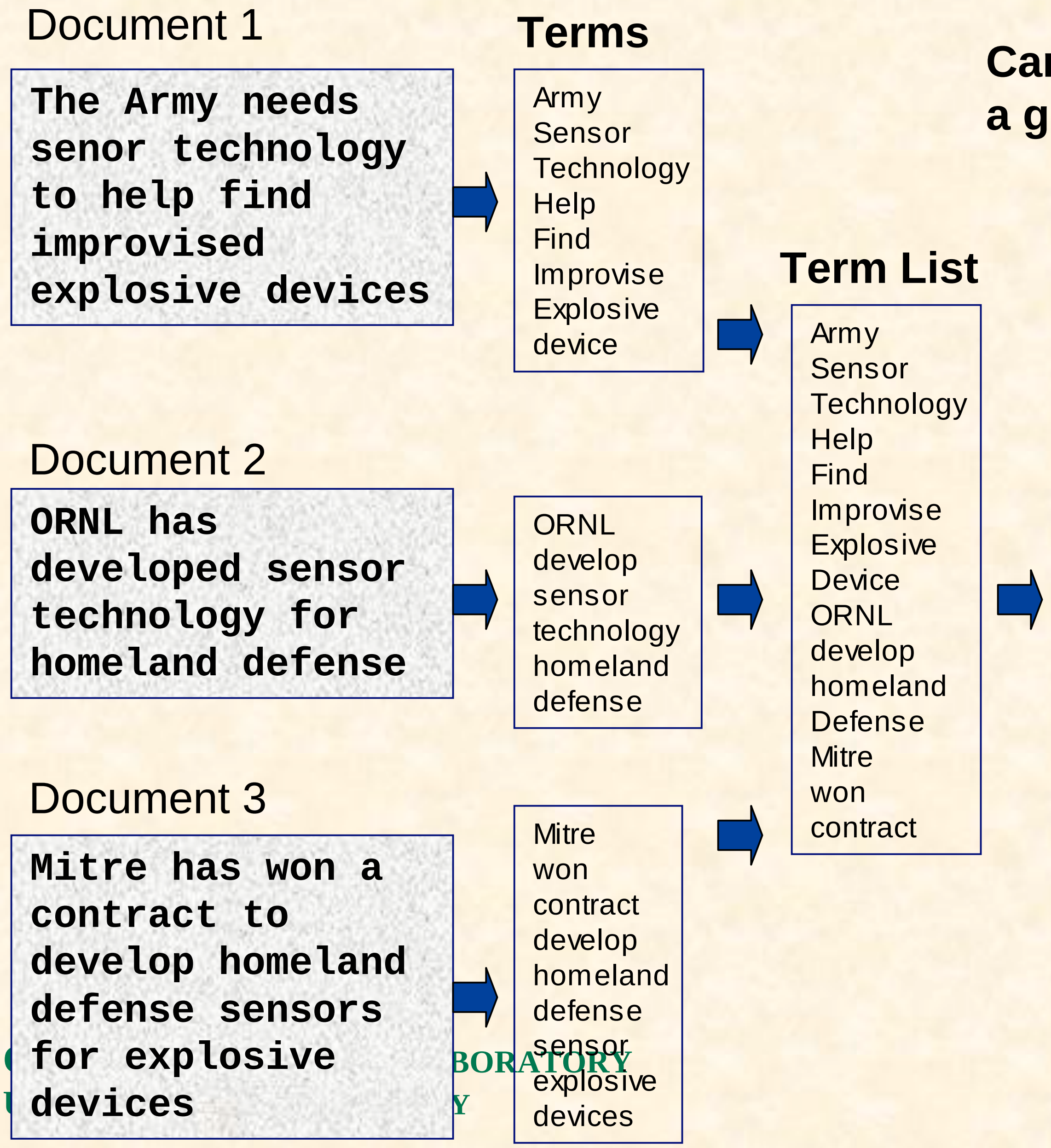
- Hundreds of pages per day
- Poor recall
- Good understanding of the meaning

- Millions of pages per day
- Perfect recall
- No understanding of the meaning

GOAL

- **Millions of pages per day**
- **Perfect recall**
- **Good understanding of the meaning**

Information Retrieval



Document 1

The Army needs senor technology to help find improvised explosive devices

Terms

Army
Sensor
Technology
Help
Find
Improvise
Explosive
device

Document 2

ORNL has developed sensor technology for homeland defense

ORNL
develop
sensor
technology
homeland
defense

Document 3

Mitre has won a contract to develop homeland defense sensors for explosive devices

Mitre
won
contract
develop
homeland
defense
sensor
explosive
devices

Term List

Army
Sensor
Technology
Help
Find
Improvise
Explosive
Device
ORNL
develop
homeland
Defense
Mitre
won
contract

Can find documents that contain a given word

Vector Space Model

	Doc 1	Doc 2	Doc 3
Army	1	0	0
Sensor	1	1	1
Technology	1	1	0
Help	1	0	0
Find	1	0	0
Improvise	1	0	0
Explosive	1	0	1
Device	1	0	1
ORNL	0	1	0
develop	0	1	1
homeland	0	1	1
Defense	0	1	1
Mitre	0	0	1
won	0	0	1
contract	0	0	1

Clustering

Vector Space Model

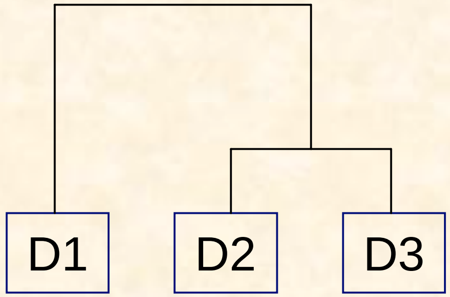
	Doc 1	Doc 2	Doc 3
Army	1	0	0
Sensor	1	1	1
Technology	1	1	0
Help	1	0	0
Find	1	0	0
Improvise	1	0	0
Explosive	1	0	1
Device	1	0	1
ORNL	0	1	0
develop	0	1	1
homeland	0	1	1
Defense	0	1	1
Mitre	0	0	1
won	0	0	1
contract	0	0	1

Dissimilarity Matrix

	Doc 1	Doc 2	Doc 3
Doc 1	100%	17%	21%
Doc 2		100%	36%
Doc 3			100%

Documents to Documents

Cluster Analysis



Most similar documents

TFIDF

$$W_{ij} = \log_2\left(f_{ij} + 1\right) * \log_2\left(\frac{N}{n}\right)$$

Euclidean distance

$$d_2(\mathbf{x}_i, \mathbf{x}_j) = \Big(\sum_{k=1}^{d} (x_{i,k} - x_{j,k})^2\Big)^{1/2}$$

Time Complexity

$$O(n^2 \mathrm{Log}\ n)$$

Limitations

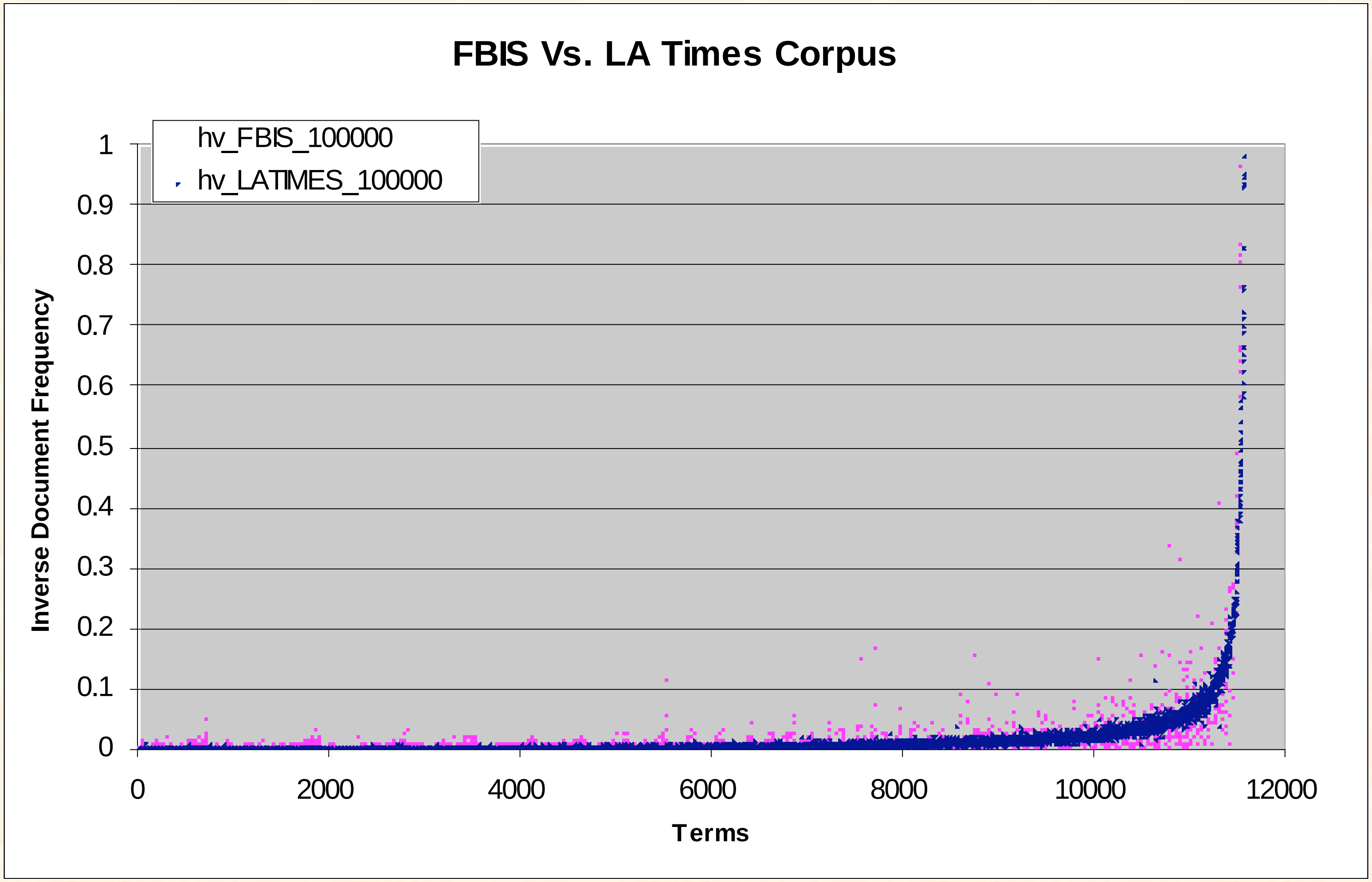
- **Term Frequency/Inverse Document Frequency (TFIDF)**

$$W_{ij} = \log_2\left(f_{ij} + 1\right) * \boxed{\log_2\left(\frac{N}{n}\right)}$$

Document Set must be known before VSM can be calculated

  - **TFIDF not practical for streaming data**
  - **Requires sequential processing**

Reference Corpus Term Frequency Distribution



TREC Text Research Collection Vol. 5.

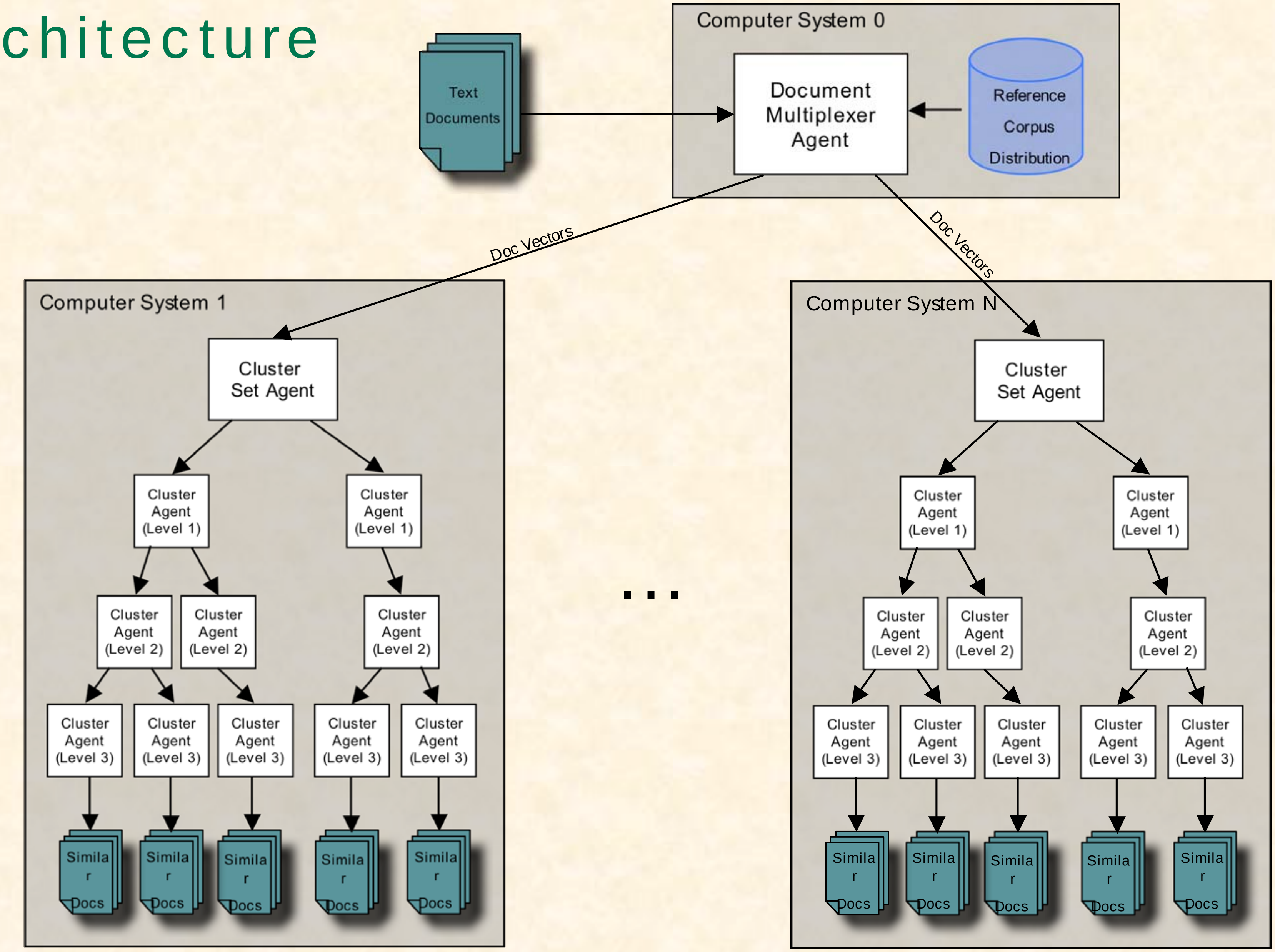
Replace IDF with reference corpus distribution

$$W_{ij} = \log_2\left(f_{ij} + 1\right) * \log_2\left(\frac{C+1}{c+1}\right)$$

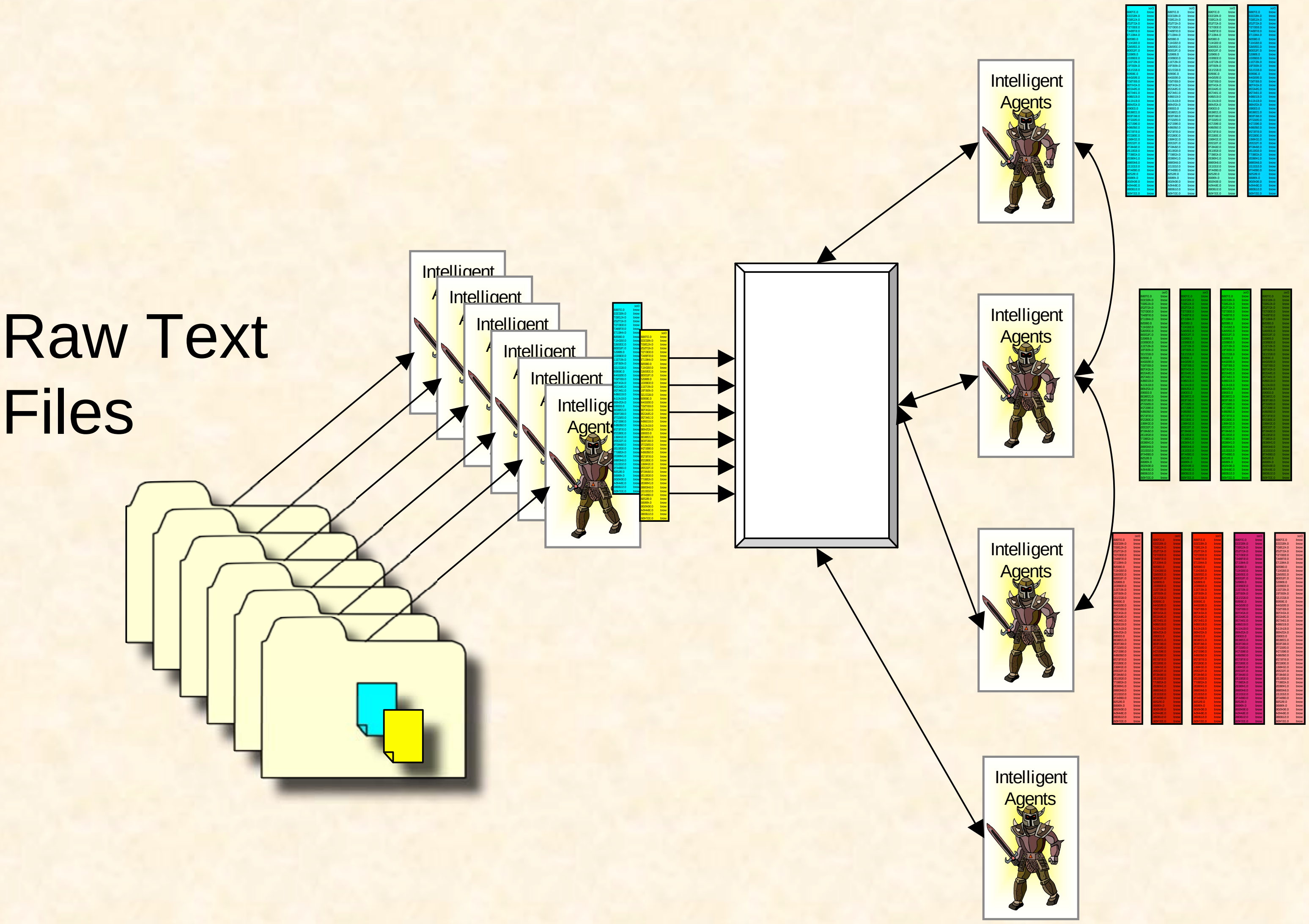
C is the number of documents our reference corpus, and c is the number of documents in the reference corpus where Tj occurs at least once.

- **The reference corpus contains 239,864 unique terms from 255,749 documents of the TREC Text Research Collection Vol. 5.**
- **Allows us to create a vector from an individually streamed document**

Architecture



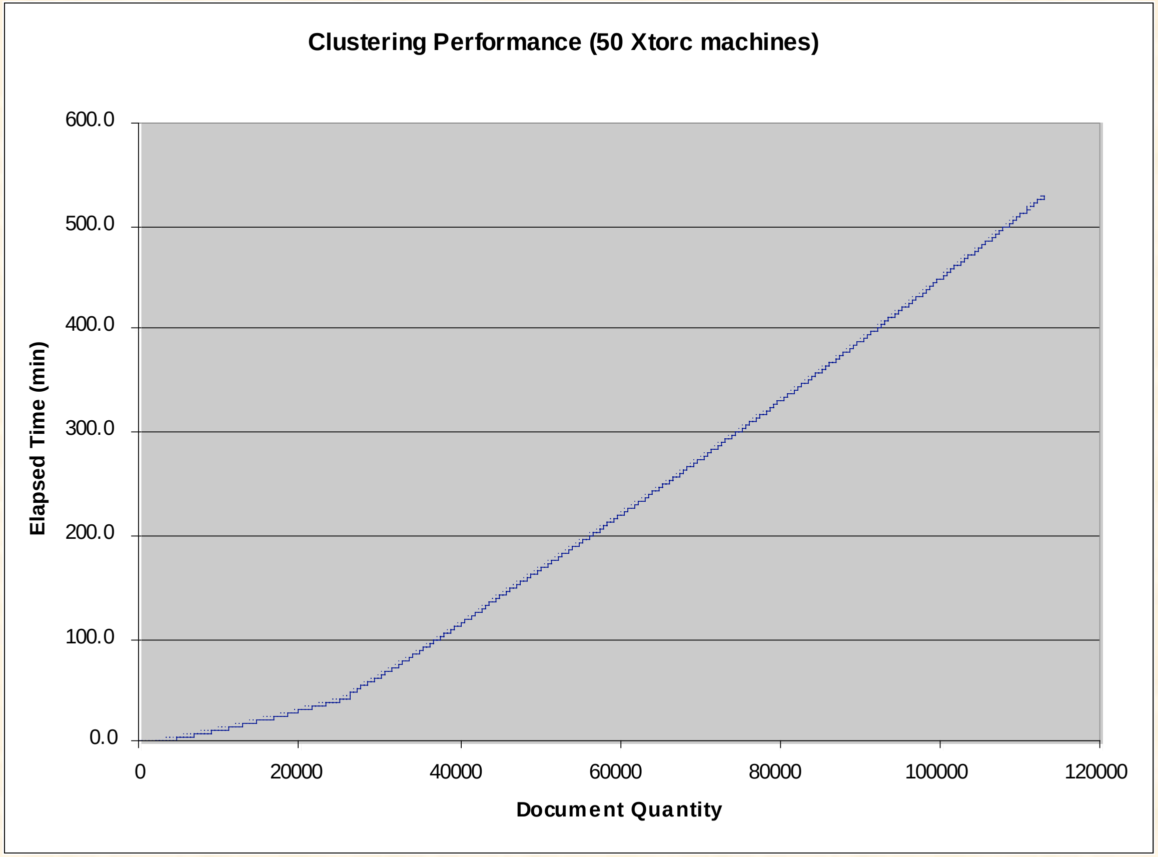
Distributed Dynamic Clustering



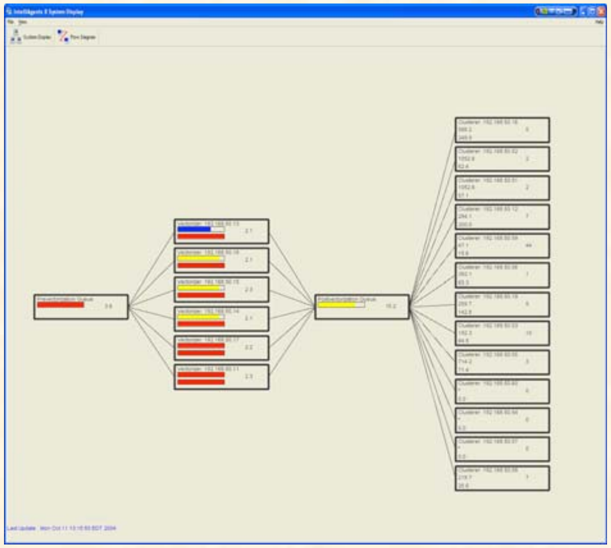
Raw Text Files

“An Agent-based Method for Distributed Clustering of Textual Information,” Patent Pending.

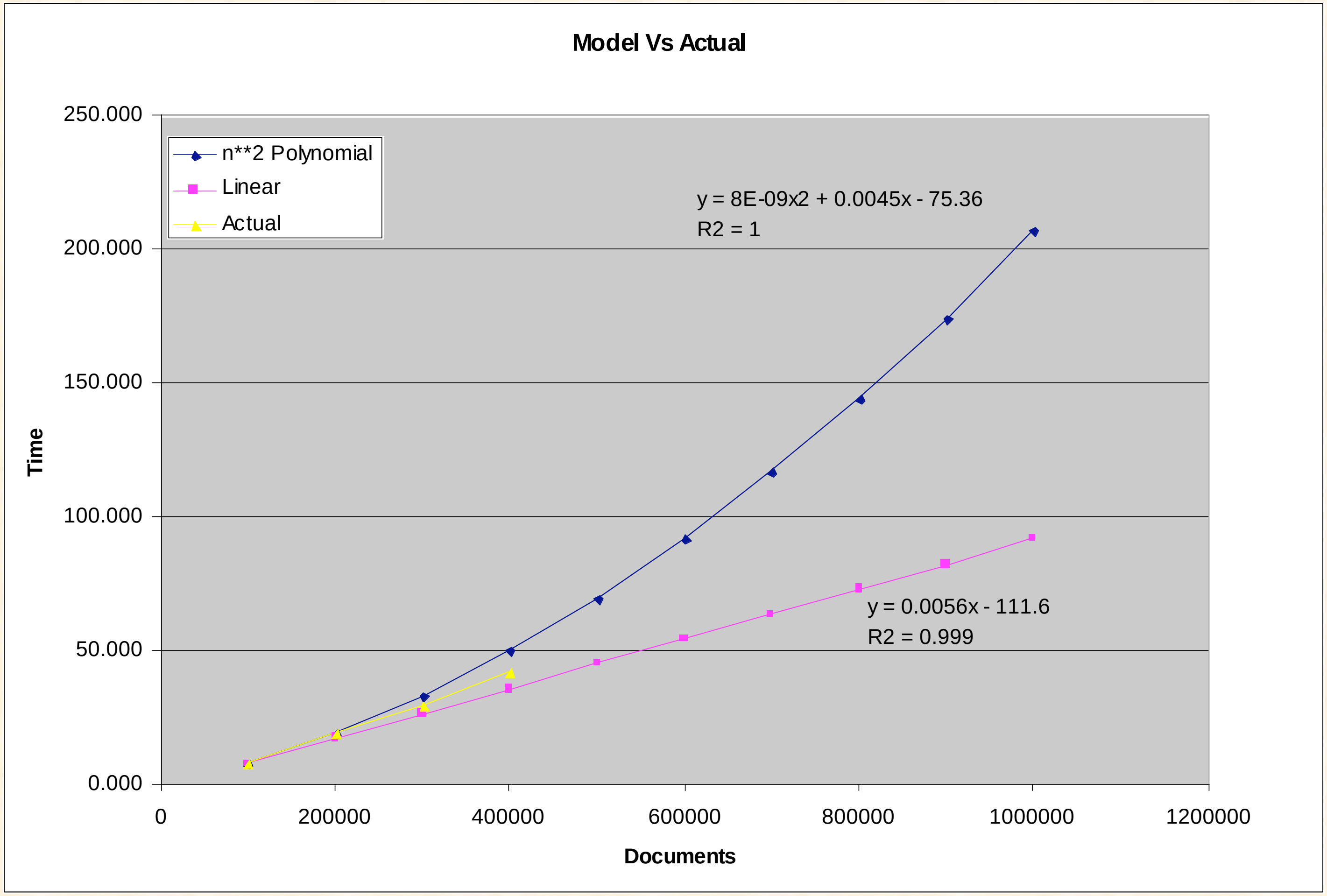
Performance Experiments



XTORC (64-node)
Pentium IV (single processor)
100/1000 Mbps Ethernet
Memory: 768KB
L2/L3 Cache: 265 KB



Initial estimates of time complexity



Next steps

- Multicast for agent communication
- Topology for agent communication
- Theoretical analysis for time and space complexity
- Cluster fidelity evaluation

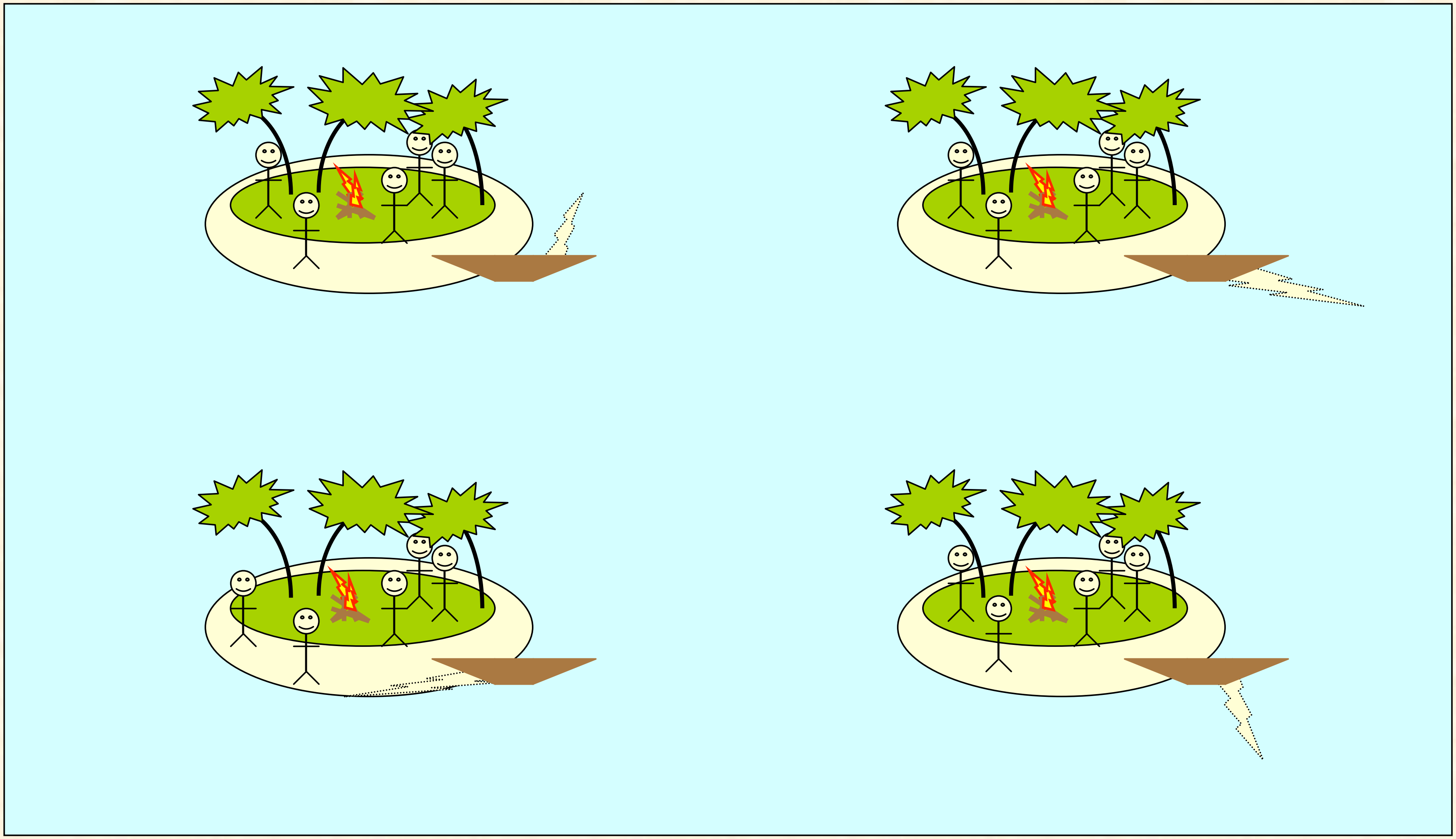
Long Term Research

- **Parallel genetic algorithms for text analysis – Dr. Robert Patton**
- **Ant swarm text analysis – Paul Palathingal**
- **Particle swarm optimization for text analysis – Dr. Xiaohui Cui**

Parallel GA – Robert Patton

- **GA can be easily parallelized and distributed**
- **Several different types of parallel GA**
- **Island model parallel GA**
  - **Islands with populations of individuals**
  - **Each island searches through a different part of the solution space**
  - **Migration of individuals occurs between islands to maintain diversity in the DNA**

Island Model GA



Ant Swarming Based Text Clustering – Paul Palathingal

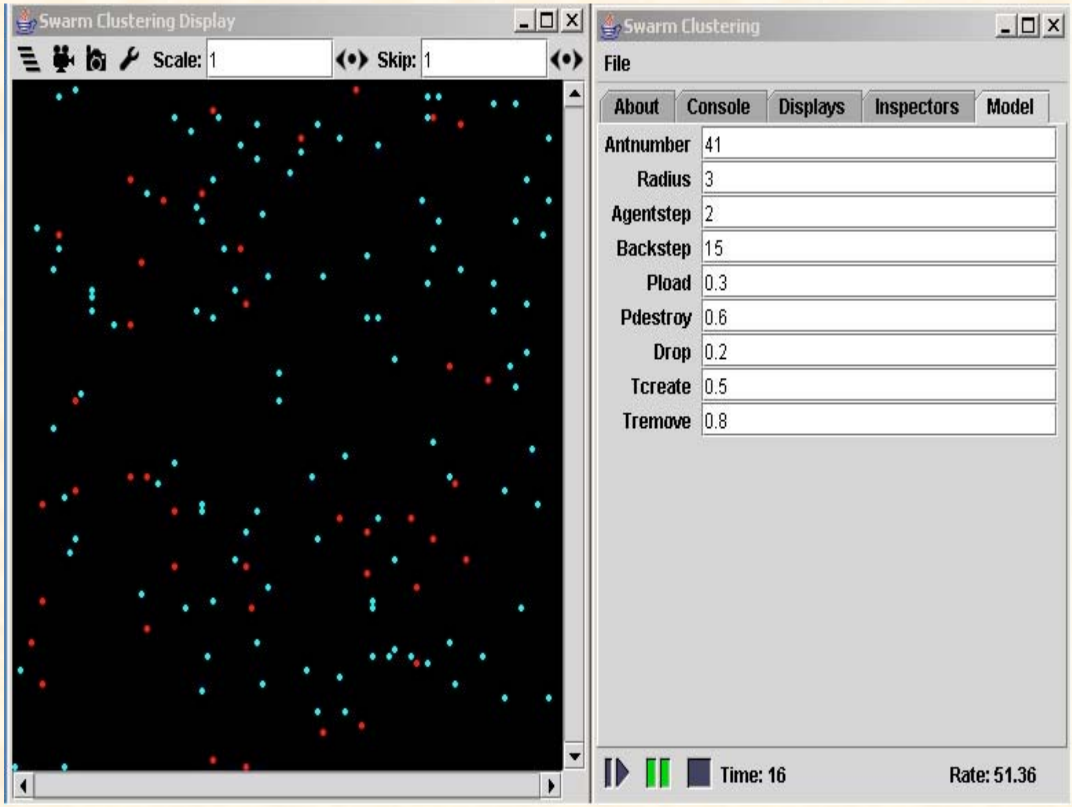
- **Introduction:**
  - **An ant is a behaviorally simple agent with limited memory**
  - **Workers in ant colonies form piles of dead ants**
  - **An item is dropped by an ant if surrounded by similar items**
  - **An item is picked up by an ant if items in the neighborhood are dissimilar**
  - **A similar approach is used towards text based clustering**

- **Approach:**
  - **Text documents are scattered randomly on a discrete 2D board**
  - **Initially the ants are randomly scattered throughout the board**
  - **The ants cluster the documents to form heaps**

Ant Swarming Based Text Clustering

Algorithm:

1. Randomly place the ants and documents on the board
2. Repeat
3. For each ant Do
   a. Move the ant
   b. If the ant does not carry a document then if there is a document in the 8 neighboring cells of the ant, the ant possibly picks up the document
   c. Else the ant looks at its 8 neighboring cells and possibly drops the document
4. Use cluster centers from above as the centers for the C Means algorithm
5. Cluster the data further using the C Means algorithm to form new heaps
6. Repeat steps 2-5 until a stopping criteria



Scale: 1

Skip: 1

Antnumber	41
Radius	3
Agentstep	2
Backstep	15
Pload	0.3
Pdestroy	0.6
Drop	0.2
Tcreate	0.5
Tremove	0.8

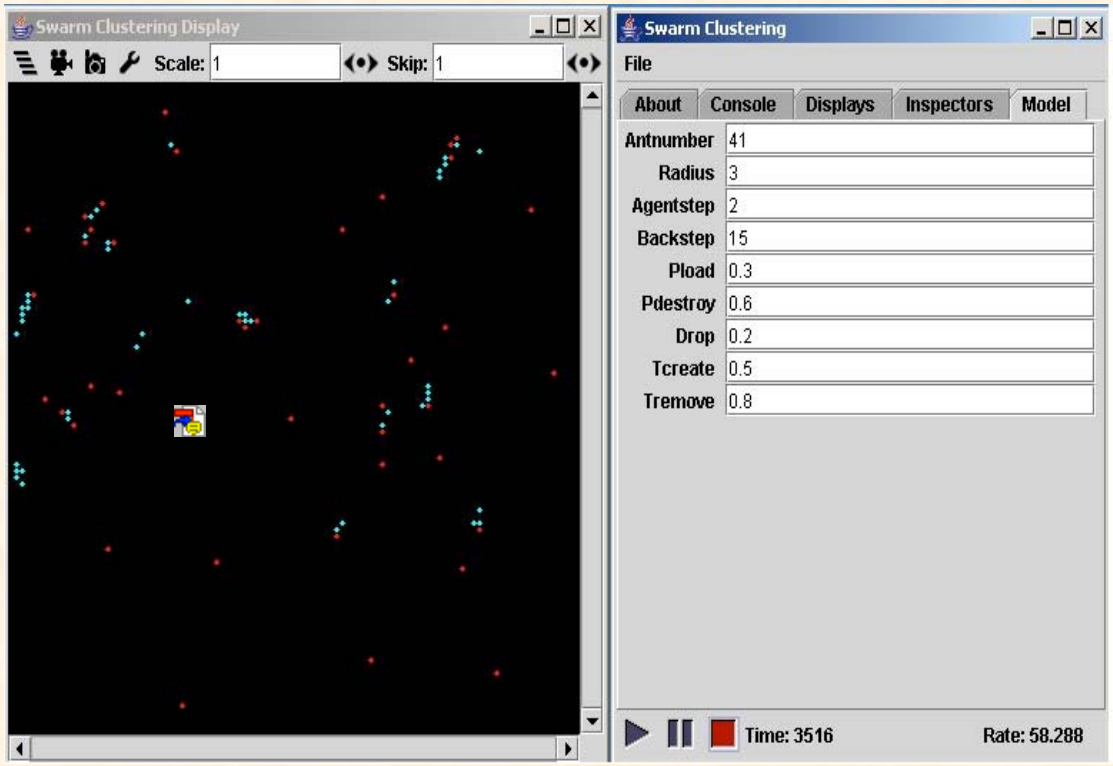
Time: 16

Rate: 51.36

Red Dot → Agent (Ant)
Blue Dot → Document

Ant Swarming Based Text Clustering Contd ..

- Improve performance by altering swarming metrics
- Incorporate the swarm algorithm into the ORMAC (Oak Ridge Mobile Agent Community) architecture
- Parallelize the code to run on a 64 node cluster computer
- Apply the clustering results towards threat document analysis and retrieval



Scale: 1
Skip: 1

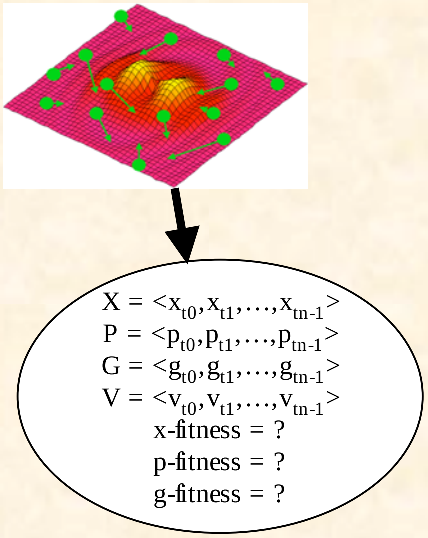
Antnumber 41
Radius 3
Agentstep 2
Backstep 15
Pload 0.3
Pdestroy 0.6
Drop 0.2
Tcreate 0.5
Tremove 0.8

Time: 3516
Rate: 58.288

Red Dot → Agent (Ant)
Blue Dot → Document

Particle Swarm Optimization (PSO) Xiaohui Cui

- ❑ **The PSO algorithm has been demonstrated as an efficient algorithm for finding the optimal solution in a stationary environment.**
- ❑ **It was introduced by Kennedy and Eberhart in 1995.**
- ❑ **A group of “particles” are thrown into the search space.**
- ❑ **Particles can be seen as simple agents that fly through the search space and record and communicate the best solutions they have discovered.**



X = <$x_{t0}, x_{t1}, \ldots, x_{tn-1}$>
P = <$p_{t0}, p_{t1}, \ldots, p_{tn-1}$>
G = <$g_{t0}, g_{t1}, \ldots, g_{tn-1}$>
V = <$v_{t0}, v_{t1}, \ldots, v_{tn-1}$>
x-fitness = ?
p-fitness = ?
g-fitness = ?

A particle (individual) is composed of:

- **Four vectors:**
  - x-vector: **particle current position**
  - v-vector: **current velocities of the particle**
  - p-vector: **location of the best solution found by the individual particle**
  - g-vector: **location of the best solution found by the whole swarm.**
- **Three fitness values:**
  - x-fitness **the fitness of the x-vector**
  - p-fitness **the fitness of the p-vector**
  - g-fitness **the fitness of the g-vector.**

$$v_d(t+1) = \alpha v_d(t) + \varphi_1 rand_1(0,1)(p_{i,d} - x_d(t)) + \varphi_2 rand_2(0,1)(p_{g,d} - x_d(t))$$

(Clerc, 2002)

$$x_d(t+1) = x_d(t) + v_d(t+1)$$

Document Clustering using PSO(1)

- Each text document can be represented using the Vector Space Model (VSM)
- the content of a document is formalized as a dot in the multi-dimension space and represented by a vector .
- A single particle in the swarm represents one possible solution for clustering the document collection.
- At each iteration, the particle adjusts the centroid vectors’ positions in the vector space according to its own experience and that of its neighbor particles.

Fitness function

- **Distance between two cluster centroid vectors m_p and m_j**

$$d(m_p, m_j) = \sqrt{\sum_{k=1}^{d_m} (m_{pk} - m_{jk})^2 / d_m}$$

  - **d_m is the space dimension.**
  - **m_{pk} and m_{jk} stand for the document m_p and m_j ’s weight values in dimension k.**

- **The fitness (evaluation) function of each particle cluster centroid vectors m_k**

$$f_e = \frac{\sum_{j=1}^{N_c} \{ \frac{\sum_{i=1}^{p} d(o_{ij}, m_j)}{p} \}}{N_c}$$

N_c	Cluster number
p	Document vector number belong to cluster C_j
m_j	Centroid vector of cluster C_j
o_{ij}	i-th document vector belong to cluster C_j

Summary

- **Main challenge to significantly improve the way text is analyzed**
- **Enhancements to TFIDF allow for parallel algorithms**
- **Agent architecture provides analysis approach that can run on cluster computers**
- **Agents provide evolutionary and swarming analysis methods**